

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平9-130259

(43)公開日 平成9年(1997)5月16日

(51)Int.Cl. <sup>4</sup>	識別記号	庁内整理番号	F I	技術表示箇所
H 0 3 M	7/30	9382-5K	H 0 3 M 7/30	B
H 0 4 N	1/41		H 0 4 N 1/41	B
	7/24		7/13	Z

審査請求 未請求 請求項の数2 O L (全 11 頁)

(21)出願番号 特願平7-283351

(22)出願日 平成7年(1995)10月31日

(71)出願人 000001443

カシオ計算機株式会社

東京都新宿区西新宿2丁目6番1号

(72)発明者 村山 洋一

東京都羽村市栄町3丁目2番1号 カシオ  
計算機株式会社羽村技術センター内

(74)代理人 弁理士 阪本 紀康

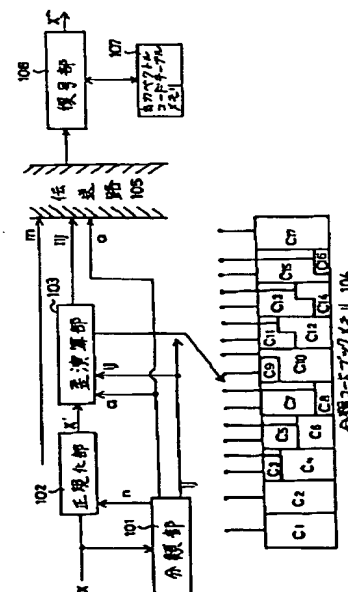
(54)【発明の名称】 ベクトル量子化符号化方式、及びベクトル復号方式

(57)【要約】

【課題】 再現画像の画質を維持しつつ、コードブックの格納に要する記憶容量を低減する。

【解決手段】 分類部101は、入力ベクトルXを分析し、それがどのクラスに属するかを判別するとともに、判別したクラスの代表ベクトルに対して行うべき変換内容を決する。これらの結果は判別情報ij、変換情報aとして歪演算部103に出力する。歪演算部103は、これらの情報に従い、正規化部102が正規化して出力した入力ベクトルX'に最も近い代表ベクトルを分類コードブックメモリ104から探索して、そのインデックスI<sub>ij</sub>を出力する。一方、復号側の復号部106は、伝送路105を介して受け取ったインデックスI<sub>ij</sub>に対応する出力ベクトルを出力ベクトルコードテーブルメモリ107から読み出し、その出力ベクトルに対して変換情報aが指定する変換内容の変換を行う。

本発明の形態が適用されたシステム構成のブロック図



## 【特許請求の範囲】

【請求項1】 予め設定されているブロックサイズでブロック化された入力ベクトルに対してベクトル量子化を行う方式であって、

前記ブロックサイズでブロック化された複数の代表ベクトルからなるコードブックを記憶したコードブック記憶手段と、

前記コードブック記憶手段に格納されている代表ベクトルを読み出すとともに、該読み出した代表ベクトルに対し、それを構成する各要素のベクトル内における配置の変換を行う代表ベクトル読出手段と、を具備し、

前記代表ベクトル読出手段が出力した前記コードブック記憶手段に記憶されている代表ベクトル、及び前記変換が行われた代表ベクトルのなかから、前記入力ベクトルに最も近い代表ベクトルを探索して、該探索した代表ベクトルを表す識別情報を符号化データとして出力する、ことを特徴とするベクトル量子化符号化方式。

【請求項2】 請求項1記載のベクトル量子化符号化方式が前記入力ベクトルに対してベクトル量子化を行った結果として出力する符号化データから、該入力ベクトルを復元する方式であって、

前記コードブック記憶手段に記憶されている複数の代表ベクトルを、それぞれ出力ベクトルとして記憶した出力ベクトル記憶手段と、

前記符号化データである識別情報に応じて前記出力ベクトル記憶手段から出力ベクトルを読み出し、該読み出した出力ベクトルに対して前記変換を行う出力ベクトル読出手段と、

を具備したことを特徴とするベクトル復号方式。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、入力信号系列を複数のサンプルにまとめてブロック化し、これを多次元空間で量子化する技術、及び量子化された入力信号系列を復号する技術に関する。

## 【0002】

【従来の技術】近年、非常に高い圧縮率を実現できるといふ優れた利点を有するベクトル量子化技術は、画像伝送等を行う際のデータ圧縮の基本技術として広く用いられている。

【0003】ベクトル量子化技術では、ベクトルの全体である信号ベクトル空間のなかから有限個の代表ベクトルを指定して表にし、符号化側、及び復号側に共通に保持させる。この表がコードブックである。入力ベクトルの符号化は、その入力ベクトルに最も近い代表ベクトルをコードブックの中から探し出し、探し出した代表ベクトルのインデックスを入力ベクトルの量子化データとして符号化することで行われる。一方の復号側においては、符号化データからインデックスを復号し、そのインデックスに対応する代表ベクトルをコードブックのなか

から探索し、その代表ベクトルを出力ベクトルとすることにより、入力ベクトルを再生する。

【0004】図8は、従来のベクトル量子化符号化装置の構成ブロック図である。この図8を参照して、従来のベクトル量子化符号化装置について具体的に説明する。図8のベクトル量子化符号化装置には、特に図示しない外部装置（画像入力装置等）から、画像の画素信号を所定数H毎にまとめてブロック化された入力ベクトルXが入力される。分類コードブックメモリ803には、図中、 $C_i$ （ $i$ は1～M（クラスの総数）までの間の整数）で示す種別（クラス或いはサブコードブック）毎に代表ベクトルを複数まとめて構成されたコードブックが格納されている。分類部801は、入力ベクトルXを分析し、この入力ベクトルXが属するクラス $C_i$ を判別する。この判別結果は、入力ベクトルXに最も近い代表ベクトルを探索するクラス $C_i$ を指定する判別情報 $i$ として、分類コードブックメモリ803から歪演算部802に出力される。

【0005】歪演算部802は、上記判別情報 $i$ に従い、分類コードブックメモリ803から探索を行う代表ベクトルのクラス $C_i$ を特定し、そのクラス $C_i$ の代表ベクトルを順次読み出して、入力ベクトルXとの歪量を算出する。クラス $C_i$ 全ての代表ベクトルを読み出し、入力ベクトルXとの歪量の算出が終了すると、そのクラス $C_i$ のなかで最も入力ベクトルXに近い（歪量が小さい）代表ベクトルのインデックス $l$ を符号化して伝送路804に送出する。

【0006】他方の受信（復号）側では、送信（符号化）側の分類コードブックメモリ803に格納されているコードブックを予め備えている。受信側（ベクトル量子化復号装置）は、伝送路804を介して送信側から送られた符号化データから、上記インデックス $l$ を復元し、そのインデックス $l$ に対応する代表ベクトルをコードブックから探索することにより、符号化データの復号を行う。このとき探索された代表ベクトルが出力ベクトルである。

## 【0007】

【発明が解決しようとする課題】上述したように、ベクトル量子化技術においては、入力ベクトルXに最も近い代表ベクトルを、予め用意したコードブックのなかから探索することで符号化を行うことを基本としている。このベクトル量子化技術では、1サンプル当たりの符号化レート、次元数（ブロックサイズ）のそれぞれの値が大きくなるのに伴い、コードブックに登録する代表ベクトル数が指数関数的に増大することが知られている。

【0008】具体的には、例えば符号化レートを一定の値としてブロックサイズだけを変更した際の1例を挙げれば、例えばウェーバの法則に基づいて代表ベクトルをクラス毎に用意すると、 $4 \times 4$ の入力ベクトルのブロック化では1クラス当たり31、 $5 \times 5$ のブロック化では

3

1クラス当たり43となる。従って、クラスの数 $M$ とすれば、コードブック全体での代表ベクトルの総数 $C$ は数1により算出される値となり、コードブックの格納に必要な記憶容量は膨大なものとなる。

【0009】

【数1】

$$C = \sum_{i=1}^M N_i$$

【0010】ここで、 $N_i$ はクラス $C_i$ に属する総代表ベクトル数である。近年、メモリ等の低価格化が進み、大容量のメモリを気軽に使用できるようになってきていることから、低符号化レートでは記憶容量の問題点はさほど大きな問題点ではなくなりつつある。しかし、この低符号化レートでは、コードブックに登録させる代表ベクトル数が比較的に少ないことから、符号化データから再現（復元）される画像の画質の劣化は大きくなるという不具合がある。他方、高符号化レートでは、高品質な画質の再現（復元）画像を得ることができ、低符号化レートが有する上記不具合は回避されるが、記憶容量の問題点は依然として深刻な問題点となっている。この問題点は、符号側だけでなく、復号側においても同様に発生している。

【0011】本発明の課題は、再現画像の画質を維持しつつ、コードブックの格納に要する記憶容量を低減することにある。

【0012】

【課題を解決するための手段】本発明のベクトル量子化符号化方式は、予め設定されているブロックサイズでブロック化された入力ベクトルに対してベクトル量子化を行うことを前提とし、ブロックサイズでブロック化された複数の代表ベクトルからなるコードブックを記憶したコードブック記憶手段と、コードブック記憶手段に格納されている代表ベクトルを読み出すとともに、該読み出した代表ベクトルに対し、それを構成する各要素のベクトル内における配置の変換を行う代表ベクトル読出手段と、を具備し、代表ベクトル読出手段が出力したコードブック記憶手段に記憶されている代表ベクトル、及び変換が行われた代表ベクトルのなかから、入力ベクトルに最も近い代表ベクトルを探索して、該探索した代表ベクトルを表す識別情報を符号化データとして出力する。

【0013】本発明のベクトル復号装置は、上記ベクトル量子化符号化方式が入力ベクトルに対してベクトル量子化を行った結果として出力する符号化データから、該入力ベクトルを復元することを前提とし、コードブック記憶手段に記憶されている複数の代表ベクトルを、それぞれ出力ベクトルとして記憶した出力ベクトル記憶手段と、符号化データである識別情報に応じて出力ベクトル記憶手段から出力ベクトルを読み出し、該読み出した出力ベクトルに対して変換を行う出力ベクトル読出手段と、を具備する。

4

【0014】代表ベクトル（出力ベクトル）を構成する各要素のブロック内における配置を変換することで、例えばその代表ベクトル（出力ベクトル）がエッジを有するものであった場合には、1つの代表ベクトル（出力ベクトル）を基にして、様々な形状のエッジを有する代表ベクトル（出力ベクトル）を新たに生成することが可能となる。新たに生成可能な代表ベクトル（出力ベクトル）をコードブックから削除しても、入力ベクトルとそれに最も近い代表ベクトル（出力ベクトル）との間の歪量は変化しないので、再現画像の画質を劣化させることなく、コードブックの記憶に必要な記憶容量は低減される。

【0015】従って、代表ベクトル（出力ベクトル）を構成する各要素のブロック内における配置を変換する手段を備えることで、符号化側、復号側の区別なく、コードメモリを記憶させるメモリの記憶容量が小さくとも、再現画像の画質を高品質に維持することが可能となる。

【0016】

【発明の実施の形態】以下、本発明の実施の形態について、図面を参照しながら詳細に説明する。図1は、本実施の形態が適用されたシステム構成のブロック図である。図1に示すシステムは、分類ベクトル量子化符号化装置、伝送路105、及びベクトル復号装置に大別される。以降の説明は、分類ベクトル量子化符号化装置、ベクトル復号装置に分けて行うことにする。

<分類ベクトル量子化符号化装置>分類ベクトル量子化符号化装置には、特に図示しない外部装置（画像入力装置等）が接続されている。この外部装置は、画像の画素信号を $K$ 個毎にまとめてブロック化し、要素 $x_1, x_2, \dots, x_K$ からなる入力ベクトル $X$ として分類ベクトル量子化符号化装置に出力する。外部装置から分類ベクトル量子化装置に出力された入力ベクトル $X$ は、分類部101、及び正規化部102に入力される。

【0017】分類部101は、例えば入力ベクトル $X$ の各隣り合う要素値（サンプル値）の1階差分をとり、その差分値に基づいてエッジを検出（差分オペレータ法）し、その検出したエッジの形状が直線状か否かにより、その入力ベクトル $X$ がエッジブロックであるか否かを判定する。入力ベクトル $X$ がエッジブロックであると判定すると、形成されているエッジの形状から、その入力ベクトル $X$ が属するクラスを判別する。

【0018】入力ベクトル $X$ がエッジブロックではないと判定、即ち、その入力ベクトル $X$ はシェイド、ミッドレンジ、混合（図2参照）といった視覚的にはっきりと認識されるエッジが含まれていない何れかのクラスに属すると判定すると、分類部101は、入力ベクトル $X$ が属するクラスを判別するための処理を行う。具体的には、例えば入力ベクトル $X$ の各要素値（サンプル値）の分散を計算し、その分散値を所定の複数のしきい値と比較することで、勾配の大きさがどのレベルに位置してい

るかを判定し、その判定結果に基づいて、入力ベクトルXが属するクラスを判別する。このようにして判別したクラスを示す判別情報 $i, j$ は、分類部101から歪演算部103に出力される。

【0019】図2は、代表ベクトルのクラス分けを説明する図である。本実施の形態では、入力ベクトルXは16次元（画像を $4 \times 4$ の画素でブロック化）である。コードブックとして登録させている代表ベクトルのブロックサイズは、この入力ベクトルXのブロックサイズと同一である。このブロックサイズのブロック（部分画像）は、図2に示すように、エッジの有無、勾配の大きさ等の模様に応じてクラス分けしている。

【0020】本実施の形態では、コードブックを $C_1 \sim C_{17}$ の計17のクラス（サブコードブック）に分類している。分類コードブックメモリ104には、各クラス別にまとめられて代表ベクトルが格納されている。各代表ベクトルは、実画像から抽出したトレーニング系列（確率分布等を考慮して抽出されたもの）を用いて、LBGアルゴリズム等の設計法に基づいて設計したものである。 $C_1 \sim C_{17}$ の各クラスにおいて、クラス $C_1$ はシェイド、クラス $C_2$ はミッドレンジ、クラス $C_3 \sim C_6$ は水平、或いは垂直方向のエッジを有するクラス、クラス $C_7 \sim C_{17}$ は対角線方向のエッジを有するクラス、クラス $C_{17}$ は混合に対応するクラスである。混合に対応するクラスは、他のクラスに属さない代表ベクトルをまとめてクラスとしたものである。

【0021】分類部101は、上述したようにクラス分けした代表ベクトルに対応させて、入力ベクトルXが属するクラスを判別し、その判別結果を判別情報 $i, j$ として歪演算部103に出力する。このとき、入力ベクトルXのクラスがミッドレンジであると判別した場合、それを示す判別情報 $i, j$ を歪演算部103に出力する他に、正規化部102に出力する判別信号 $n$ をアクティブにする。

【0022】正規化部102は、入力ベクトルXのベクトル内平均値 $m$ を求め、入力ベクトルXを構成する各要素 $x_i$ を、該求めた平均値 $m$ を用いて平均値分離正規化し、正規化後の入力ベクトルXを入力ベクトル $X'$ として歪演算部103に出力する。上記平均値 $m$ は、符号化されて伝送路105に出力される。

【0023】また、正規化部102は、分類部101から入力した判別信号 $n$ がアクティブになると、正規化後の入力ベクトルXに対してDCT演算を行い、各要素 $x_i$ の値をDCT係数に変換する。その後、これらDCT係数を入力ベクトル $X'$ として歪演算部103に出力する。

【0024】上述したように、判別信号 $n$ は、分類部101で入力ベクトルXはミッドレンジのクラスに属すると判別した場合にアクティブになる。このため、ミッドレンジのクラスとして分類コードブックメモリ104に

格納されている代表ベクトルは、DCT変換されたトレーニング系列を用いて、LBGアルゴリズム等の設計法に基づいて設計されている。なお、信号信号 $n$ がインアクティブであれば、正規化部102は、正規化後の入力ベクトルXをそのまま入力ベクトル $X'$ として歪演算部103に出力する。

【0025】歪演算部103は、分類部101から出力された判別情報 $i, j$ に従い、それが指定するクラス $C_{ij}$ に属する代表ベクトルを分類コードブックメモリ104から順次読み出し、読み出した代表ベクトルと正規化部102から入力した入力ベクトル $X'$ との間の歪量を算出する。指定されたクラス $C_{ij}$ に属する全ての代表ベクトルを読み出して、入力ベクトル $X'$ との歪量をそれぞれ算出すると、そのクラス $C_{ij}$ 内で入力ベクトル $X'$ に最も近い代表ベクトルを示すインデックス $l_{ij}$ を符号化して伝送路105に出力する。

【0026】分類部101は、歪演算部103に対し、判別情報 $i, j$ の他に、変換情報 $a$ を出力する。この変換情報 $a$ は、ブロック内における各要素値の配置を変換する内容を示すものである。ここで、図3～図5を参照して、上記変換について説明する。

【0027】図3は、エッジブロックの変換方法の説明図である。この図3において、斜線で示す部分がエッジに対応しており、左側のブロックが変換前のもの（オリジナル）である。

【0028】オリジナルのブロックの右側に示す上下2つのブロックは、オリジナルのブロックに対して、ブロックの中心を軸にして90度の回転変換（左回りが正である）を行った際の状態（上段）、ブロックの中心を水平方向に横切る線を仮定して鏡像変換を行った際の状態（下段）をそれぞれ示している。

【0029】上記90度の回転変換を行うと、例えばオリジナルのブロック内において左上隅の要素は左下隅にブロック内における配置が変換される。他の各要素も同様にして、ブロック（ベクトル）内における配置が変換される。一方、ブロックの中心を水平方向に横切る線を仮定して鏡像変換を行うと、その線により上部分、下部分に分割された各部分は、その線に対して対称に配置し直される。これにより、例えばオリジナルのブロック内において左上隅の要素は左下隅に、右上隅の要素は右下隅に、ブロック内における配置が変換される。これは、分類コードブックメモリ104からエッジブロックを読み出すときに、オリジナルブロックの読み出しアドレスを変えることにより実現できる。すなわち、オリジナルブロックを左上から右に順次アドレスを変えて読み出せば、オリジナルブロックが読み出せる。これに対して、90°左回転の場合は、右上から下に順次アドレスを変えて読み出せばよい。また、鏡像変換の場合は、左下から右に順次アドレスを変えて読み出せばよい。

【0030】図3は、エッジブロックに対して90度の

回転変換、水平方向の線を仮定した鏡像変換を行った例だけを示しているが、当然のことながら、変換方法はこれに限定されない。これらの他に、180度、270度(−90度)の回転変換、垂直方向の線を仮定した鏡像変換等がある。

【0031】本実施の形態では、上述したような配置変換を行うことにより、1つの代表ベクトル(エッジブロック)を、そのエッジの形状とは異なる他の代表ベクトル(エッジブロック)の代替として用いている。図4、及び5を参照して、代表ベクトル(エッジベクトル)の代替方法について具体的に説明する。

【0032】図4は、水平方向、或いは垂直方向にエッジを有する基本的なエッジブロックを示している。この図4において、(A)としてまとめた(a1)~(a3)の3つのエッジブロックが、各クラスC<sub>1</sub>~C<sub>3</sub>に属する代表ブロックによって表現されるものである。

【0033】図4では、上記(A)の他に、(B)~(D)にそれぞれ3つのエッジブロックをまとめて示している。これら(B)としてまとめた(b1)~(b3)、(C)としてまとめた(c1)~(c3)、及び(D)としてまとめた(d1)~(d3)は、(A)としてまとめた(a1)~(a3)の3つのエッジブロックに対し、その要素を配置変換することにより表現されたものである。

【0034】具体的には、(a1)のエッジブロックに対し、90度の回転変換を行うことにより(c1)、−90度(270度)の回転変換を行うことにより(d3)のエッジブロックがそれぞれ得られ、また、水平方向の線を仮定した鏡像変換を行うことにより(b3)のエッジブロックが得られる。他のオリジナルのエッジブロックにおいては、(a2)のエッジブロックに対し、回転変換を行うことにより(c2)及び(d2)のエッジブロック、水平方向の線を仮定した鏡像変換を行うことにより(b2)のエッジブロックが得られる。(a3)のエッジブロックに対しては、回転変換を行うことにより(c3)及び(d1)のエッジブロック、水平方向の線を仮定した鏡像変換を行うことにより(b1)のエッジブロックが得られる。

【0035】図5は、対角線(45度、或いは135度)方向のエッジを有する基本的なエッジブロックを示している。この図5において、(A)としてまとめた(a1)~(a4)の4つのエッジブロックが、各クラスC<sub>1</sub>~C<sub>4</sub>に属する代表ブロックによって表現されるものである。

【0036】図5では、上記(A)の他に、(B)~(D)にそれぞれ4つのエッジブロックをまとめて示している。これら(B)としてまとめた(b1)~(b4)、(C)としてまとめた(c1)~(c4)、及び(D)としてまとめた(d1)~(d4)は、(A)としてまとめた(a1)~(a4)の4つのエッジブロッ

クに対し、その要素を配置変換することにより表現されたものである。

【0037】具体的には、(a1)のエッジブロックに対して180度、−90度(270度)、90度の回転変換をそれぞれ行うことにより、(b4)、(c1)、及び(d4)のエッジブロックが得られる。他のオリジナルのエッジブロックにおいては、(a2)のエッジブロックに対してそれぞれ異なる角度の回転変換を行うことにより、(b3)、(c2)、及び(d3)のエッジブロックが得られる。(a3)のエッジブロックに対してそれぞれ異なる角度の回転変換を行うことにより、(b2)、(c3)、及び(d2)のエッジブロックが得られる。(a4)のエッジブロックにおいては、それに対してそれぞれ異なる角度の回転変換を行うことにより、(b1)、(c4)、及び(d1)のエッジブロックが得られる。

【0038】このように、1つの代表ベクトル(エッジブロック)を基にして、それを構成する各要素の配置を変換することにより、他のエッジが異なる代表ベクトル(エッジベクトル)を新たに生成することができる。このため、新たに生成可能な代表ベクトルを分類コードブックメモリ104に格納させておく必要が回避され、コードブックの格納に要する記憶容量を低減させることができる。この結果、分類コードブックメモリ104の記憶容量が小さくとも、再現(復元)画像の画質を高品質に維持することができる。このことを言い換えれば、同じ記憶容量のコードブックの記憶用メモリを用いた場合、本発明は従来例と比較して、高品質な再現画像が得られることを意味する。

【0039】なお、本実施の形態では、上記変換を行うブロック(代表ベクトル)をエッジブロックだけとしているが、変換を行うブロックはエッジ等の特定の模様を有するブロックに限定するものではなく、例えばシェイド、ミッドレンジ等の視覚的にエッジがはっきりしていないブロックに対しても上記と同様な変換を行って良いものである。本実施の形態において、変換を行うブロックをエッジブロックとしたのは、そのエッジの形状により、オリジナルのエッジブロックに対して変換を行うことで得られる他のエッジブロックを視覚的に明確に示すことができることから、変換を行うことで得られる効果を理解させるのが容易であると判断したためである。幅広い種類のオリジナルのブロックに対する変換を行うようにすることにより、コードブックに登録する代表ベクトル数をより少なくすることができるが、これによる再生画像の画質の劣化も回避することができる。

【0040】分類部101は、入力ベクトルXを分析してそれが水平方向、垂直方向、或いは対角線方向に直線状にエッジが形成されているエッジブロックであると判定した場合、入力ベクトルXで表現されるエッジの形状と、コードブックに登録されている各クラスの代表ベク

トルで表現されるエッジの形状とを比較して、入力ベクトルXが属するクラスを示す判別情報ij、及びその判別情報ijによって指定するクラスの代表ベクトルに対する変換内容を決定し、その変換内容を示す変換情報aを歪演算部103に出力する。この変換情報aは、歪演算部103の他に、伝送路105に出力される。

【0041】図6は、歪演算部103の構成ブロック図である。同図を参照して、歪演算部103について詳細に説明する。正規化部102から入力した入力ベクトルX'は、それを構成する各要素(x<sub>1</sub>、x<sub>2</sub>、・・・、x<sub>n</sub>)に分割されて入力ベクトルレジスタ601に保持される。入力ベクトルレジスタ601は、入力ベクトルX'に最も近い代表ベクトルが確定するまでの間、その入力ベクトルX'を保持する。

【0042】コードテーブルアドレスカウンタ602は、分類コードブックメモリ104及びインデックスラッチ608に対し、1～N<sub>i1</sub>(N<sub>i1</sub>はクラスC<sub>i1</sub>に属する総代表ベクトル数)まで変化するインデックスkを順次出力する。

【0043】分類コードブックメモリ104から読み出される代表ベクトルy<sub>k</sub>は、分類部101が出力する判別情報ij、及び上記コードテーブルアドレスカウンタ602が出力するインデックスkにより制御される。具体的には、例えば以下のようにして制御される。

【0044】判別情報ijは、歪演算部103の他に、分類コードブックメモリ104に出力される。分類コードブックメモリ104は、この判別情報ijをデコードし、判別情報ijが指定したクラスC<sub>i1</sub>の代表ベクトルy<sub>k</sub>が格納されている領域を特定する。他方のコードテーブルアドレスカウンタ602が順次出力するインデックスkは、判別情報C<sub>i1</sub>が指定したクラスC<sub>i1</sub>内における代表ベクトルkの相対的なインデックスである。分類コードブックメモリ104は、判別情報ijの他に、コードテーブルアドレスカウンタ602が出力したインデックスkを入力することで、読み出すべき代表ベクトルy<sub>k</sub>(クラスC<sub>i1</sub>内におけるインデックスがkの代表ベクトル)を特定し、それを歪演算部103のコードブック出力ベクトルレジスタ603に出力する。

【0045】このようにして、分類コードブックメモリ104が読み出す代表ベクトルy<sub>k</sub>は、判別情報ij、及びインデックスkにより制御される。以降、コードテーブルアドレスカウンタ602が出力するインデックスkについては、他と区別するために相対インデックスkと記載することにする。

【0046】コードブック出力ベクトルレジスタ603は、分類コードブックメモリ104から出力された代表ベクトルy<sub>k</sub>を、それを構成する各要素(y<sub>k1</sub>、y<sub>k2</sub>、・・・、y<sub>kn</sub>)に分割して保持する。このコードブック出力ベクトルレジスタ603には、分類部101が出力した変換情報aが入力されている。コードブック出力ベ

クトルレジスタ603は、その変換情報aが示す代表ベクトルy<sub>k</sub>に対する変換内容に従い、代表ベクトルy<sub>k</sub>を構成する各要素の並びを変更して保持する。これにより、コードブック出力ベクトルレジスタ603に、回転変換、鏡像変換等が施された状態で代表ベクトルy<sub>k</sub>の各要素が保持されることになる。

【0047】並列減算器604は、入力ベクトルレジスタ601に保持されている入力ベクトルX'の各要素と、コードブック出力ベクトルレジスタ603に保持されている代表ベクトルy<sub>k</sub>の各要素とをそれぞれ対応させて、各対応する要素間の差を求める減算を並列で行う。これにより、入力ベクトルX'の要素x<sub>1</sub>と代表ベクトルy<sub>k</sub>の要素y<sub>k1</sub>との差(=x<sub>1</sub>-y<sub>k1</sub>、或いはその逆)、要素x<sub>2</sub>と要素y<sub>k2</sub>との差、・・・、及び要素x<sub>n</sub>と要素y<sub>kn</sub>との差がそれぞれ算出される。これら各対応する要素間の減算結果は、並列絶対値演算器605にそれぞれ出力される。

【0048】並列絶対値演算器605は、並列減算器604から入力した各減算結果に対し、その絶対値を求める演算を並列で行う。これにより、並列減算器604が出力した各減算結果はその符号が統一されて、並列絶対値演算器605から絶対値歪検出器606に出力される。

【0049】絶対値歪検出器606は、並列絶対値演算器605から符号が正に統一された各要素間の減算結果を入力すると、入力ベクトルX'とコードブック出力ベクトルレジスタ603に保持されている代表ベクトルy<sub>k</sub>との間の歪量d<sub>k</sub>を算出する。この歪量d<sub>k</sub>は、例えば数2により算出される値である。絶対値歪検出器606が算出した歪量d<sub>k</sub>は、最小歪出力ベクトル検出器607に出力される。

【0050】

【数2】

$$d_k = \sum_{i=1}^n |x_i - y_{ki}|$$

【0051】最小歪出力ベクトル検出器607は、入力ベクトルX'が入力ベクトルレジスタ601に保持されている間、即ち判別情報ijが指定したクラスC<sub>i1</sub>内で入力ベクトルX'に最も近い代表ベクトルy<sub>k</sub>を確定するまでの間、絶対値歪検出器606が出力した歪量d<sub>k</sub>を保持しており、絶対値歪検出器606から歪量d<sub>k</sub>を新たに入力する度に、この新たに入力した歪量d<sub>k</sub>とそれまでに入力した他の歪量d(d<sub>1</sub>～d<sub>k-1</sub>)とを比較して、新たに入力した歪量d<sub>k</sub>がこれまでの歪量dのなかで最小か否かを判定する。最小歪出力ベクトル検出器607は、新たに入力した歪量d<sub>k</sub>がこれまでの歪量dのなかで最小であると判定した場合にのみ、インデックスラッチ608に出力する信号をアクティブにする。

【0052】インデックスラッチ608は、最小歪出力ベクトル検出器607からの出力がアクティブになる

11

と、コードテーブルアドレスカウンタ602が出力している相対インデックスkをラッチし、最小歪出力ベクトル検出器607からの出力が再びアクティブになるまでそれを保持する。

【0053】上記最小歪出力ベクトル検出器607は、特に図示していないが、インデックスラッチ608に出力する信号をアクティブにするタイミングから所定のタイミングだけ遅らせて、コードテーブルアドレスカウンタ602に、分類コードブックメモリ104から次の代表ベクトル $y_{i,j}$ の読み出しを促す信号を出力する。この信号が最小歪出力ベクトル検出器607からコード

テーブルアドレスカウンタ602に出力されることで、入力ベクトル $X'$ に最も近い代表ベクトル $y$ を探す探索が進行する。

【0054】インデックスラッチ608に保持されている相対インデックスkはクラス $C_{i,j}$ における相対的なインデックスである。このため、インデックス加算器609は、クラス $C_{i,j}$ 内における相対インデックスkが対応するコードブック全体における絶対的なインデックス $I_{i,j}$ を算出する。このインデックス $I_{i,j}$ は、例えば数3に

【0055】

【数3】

$$I_{ij} = k + \sum_{l=1}^{C-1} N_l$$

【0056】インデックス加算器609は、インデックス $I_{i,j}$ を算出すると、このインデックス $I_{i,j}$ を、入力ベクトル $X$ に最も近い代表ベクトル $y$ を示す情報として、伝送路105に出力する。

【0057】以上が、歪演算部103の構成、及び各部の動作の詳細である。次に、図7を参照して、分類部101の処理動作について詳細に説明する。図7は、分類処理の動作フローチャートである。この動作フローチャートは、分類部101が、1入力ベクトル $X$ を入力してからそれが属するクラスの判別を終了するまでの一連の処理動作を表現したものである。

【0058】まず、ステップ701では、入力ベクトル $X$ を入力する。続くステップ702では、入力した入力ベクトル $X$ を分析して、その入力ベクトル $X$ が表現する部分画像（ブロック）の模様の特徴を抽出し、入力ベ

クトル $X$ が属するクラスの判別を行う。

【0059】上記入力ベクトル $X$ の分析は、上述したように、入力ベクトル $X$ の各要素 $x_i$ の分散値、差分オペレータ法等を用いて検出したエッジの形状等に基づいて行われる。

【0060】ステップ702に続くステップ703では、判別信号 $n$ を正規化部102に出力する。このとき出力される判別信号 $n$ は、上記ステップ702で入力ベクトル $X$ がミッドレンジに対応するクラスであると判別した場合にはアクティブ、そうでないと判別した場合に

12

はイナクティブである。判別信号 $n$ を出力すると、ステップ704の処理に移行する。

【0061】ステップ704では、入力ベクトル $X$ はエッジを有するか否かを判定する。入力ベクトル $X$ で表現される部分画像内に、水平、垂直、対角線の何れかの方向にエッジが形成されていた場合、その判定はYESとなってステップ705の処理に移行する。そうでない場合、ステップ707の処理に移行する。

【0062】本実施の形態では、上述したように、エッジブロックである代表ベクトルは基本とするものだけをコードブックに登録させ、代表ベクトルを構成する各要素 $y_i$ のベクトル内における配置を変換することにより、様々なエッジブロックを生成させている。このため、ステップ704でYES、即ち入力ベクトル $X$ はエッジブロックであると判定すると、続くステップ705において、基本となる代表ベクトル $y_i$ のエッジの形状、入力ベクトル $X$ のエッジの形状から、基本となる代表ベクトル $y_i$ に対し、入力ベクトル $X$ のエッジの形状に合わせるために行うべき変換内容を決定する。

【0063】ステップ705に続くステップ706では、ステップ705で決定した変換内容を示す変換情報 $a$ を歪演算部103、及び伝送路105に出力する。変換情報 $a$ を出力すると、ステップ707の処理に移行する。

【0064】ステップ707では、入力ベクトル $X$ が属すると判別したクラスを示す判別情報 $i, j$ を歪演算部103に出力する。この判別情報 $i, j$ を出力することで、1入力ベクトル $X$ のクラスの判別に関わる一連の処理動作を終了する。一連の処理動作を終了した後は、歪演算部103が伝送路105にインデックス $I_{i,j}$ を出力し、符号化が完了するのを待つ待機状態に移行する。

【0065】なお、本実施の形態では、インデックス $I_{i,j}$ の他に、そのインデックス $I_{i,j}$ に対応する代表ベクトルに対して行うべき変換内容を示す変換情報 $a$ を伝送路105に出力しており、これらインデックス $I_{i,j}$ 、及び変換情報 $a$ が入力ベクトル $X$ に最も近い代表ベクトルを示す識別情報に対応しているが、例えば変換内容別に分類したクラスを新たに追加し、変換情報 $a$ をインデックス $I_{i,j}$ に反映させるようにすることにより、変換情報 $a$ の伝送路 $a$ への出力を不要にすることもできる。このようにした場合には、インデックス $I_{i,j}$ のみが上記識別情報に対応することになる。

【0066】新たに追加したクラスにおいては、それに属する代表ベクトルは他のクラスに属する代表ベクトルに全て代替させることができる。また、同一のクラス内であっても、1つの代表ベクトルを他の代表ベクトルの代替とすることができる。この後者においては、代替される代表ベクトルに対してインデックスを割り当てても良い。この場合にも、その代替される代表ベクトルに割り当てたインデックスは、本実施の形態におけるインデ

ックス $I_{11}$ に変換情報 $a$ を反映させた識別情報に対応するものとなる。

【0067】また、本実施の形態は、分類型のベクトル量子化符号化装置に本発明を適用させたものであるが、本発明はこれに限定するものではない。代表ベクトルをクラス分けしていない一般のもの、入力ベクトル（代表ベクトル）のブロックサイズを複数のブロックサイズのなかから選択できる階層型といった装置の形態に関わらず、本発明は適用できるものである。また、本発明はベクトル量子化の対象として画像を特に意識しているが、本発明の適用は画像のベクトル量子化にのみ限定されるものではない。音声等の他の対象に対しても適用が可能である。

＜ベクトル復号装置＞図1に示すベクトル復号装置は、伝送路105を介して分類ベクトル量子化符号化装置が送出した符号化データを受信する復号部106と、送信側のコードブックに登録されている代表ベクトルを出力ベクトルとして全て格納している出力ベクトルコードテーブルメモリ107とを備えて構成される。

【0068】送信側からは、符号化データとして、入力ベクトル $X$ 内の各要素 $x_i$ の平均値 $m$ 、インデックス $I_{11}$ 、及び変換情報 $a$ が送られる。復号部106は、伝送路105から受け取った符号化データからこれらを復号した後、インデックス $I_{11}$ に対応する出力ベクトルを出力ベクトルコードテーブルメモリ107から読み出し、該読み出した出力ベクトルに対して変換情報 $a$ 、平均値 $m$ に基づいた処理を行った後、出力ベクトル $X^*$ としてそれを出力する。

【0069】変換情報 $a$ 、平均値 $m$ に基づいた出力ベクトルに対する処理は、例えば以下に行われる。変換情報 $a$ が出力ベクトルに対して何らかの変換を行うことを指示している場合、そうでない場合に分けて説明する。

【0070】まず、変換情報 $a$ が出力ベクトルに対する何らかの変換を行うことを指示している場合について説明する。この場合、復号部106は、変換情報 $a$ の指示に従い、出力ベクトルコードテーブルメモリ107から読み出した出力ベクトルの各要素の配置を変換した後、平均値 $m$ を用いて各要素の値を修正する。この修正後の出力ベクトルが出力ベクトル $X^*$ として出力される。

【0071】一方、変換情報 $a$ が出力ベクトルに対する何らかの変換を行うことを指示していない場合、復号部106は、インデックス $I_{11}$ で指定された出力ベクトルがミッドレンジに対応するクラスであるか否か判定し、その出力ベクトルがミッドレンジに対応するクラスであると判定すると、出力ベクトルに対して逆DCT演算を行う。出力ベクトルがミッドレンジに対応するクラスではないと判定すると、出力ベクトルに対する逆DCT演算は行わない。その後、平均値 $m$ を用いて各要素の値を修正して、出力ベクトル $X^*$ を出力する。

【0072】上記したようにして復号部106は、インデックス $I_{11}$ に対応する出力ベクトルを出力ベクトルコードテーブルメモリ107から読み出し、該読み出した出力ベクトルに対して変換情報 $a$ が指示する変換内容に応じた処理を行う。このため、送信（符号化）側が送信した符号化データから、送信側が符号化した画像を忠実に再現することができる。この受信側においても、送信側と同様に、出力ベクトルコードテーブルメモリ107の記憶容量が小さくとも高品質な再現画像が得られるという効果がある。なお、本実施の形態のベクトル復号装置は、送信側から符号化データとして送出された変換情報 $a$ に従って出力ベクトルの変換を行っているが、本発明はこの復号装置においても様々な変形が可能である。例えば送信側がインデックスのみを送出する場合には、代替させる出力ベクトルとインデックスの対応関係、及び代替させる出力ベクトルを得るための生成情報（基となる出力ベクトルのインデックス、この出力ベクトルに対する変換内容等）等を予め用意しておくことで対応することができる。このことは、本発明のベクトル量子化符号化方式、及びベクトル復号方式は、共にそれぞれを独立させて適用させることも可能であることを意味している。

【0073】

【発明の効果】以上、説明したように本発明のベクトル量子化符号化方式は、コードブックに登録した代表ベクトルを基にして、代表ベクトルを構成する各要素のブロック内における配置を変換して他の代表ベクトルを新たに生成する。このため、少ない代表ベクトルを登録したコードブックであっても、入力ベクトルとの歪量が小さい代表ベクトルを探索することができ、再現画像の画質を高品質に維持することができる。

【0074】他方の本発明のベクトル復号方式においても、コードブックに登録した出力ベクトルを基にして、出力ベクトルを構成する各要素のベクトル内における配置を変換して他の出力ベクトルを新たに生成するため、上記本発明のベクトル量子化符号化方式と同様の効果を得ることができる。

【図面の簡単な説明】

【図1】本実施の形態が適用されたシステム構成のブロック図である。

【図2】ブロックのクラス分けを説明する図である。

【図3】エッジブロックの変換方法の説明図である。

【図4】エッジブロックを示す図である（その1）。

【図5】エッジブロックを示す図である（その2）。

【図6】歪演算部103の構成ブロック図である。

【図7】分類処理の動作フローチャートである。

【図8】従来のベクトル量子化符号化装置のブロック図である。

【符号の説明】

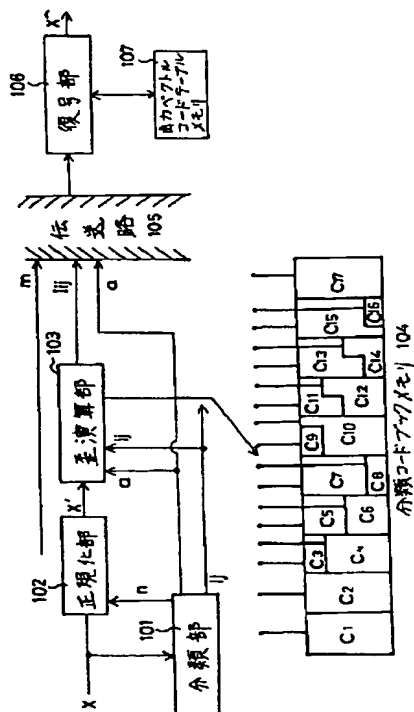
50 101 分類部



103 歪演算部  
104 分類コードブックメモリ  
106 復号部

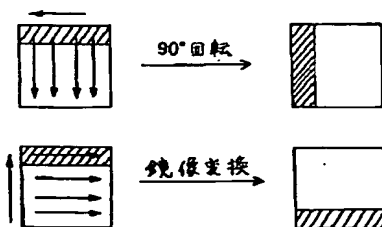
【図1】

本実施の形態で適用されたシステム構成のブロック図



【図3】

エッジブロックの変換方法の説明図



(9)

特開平9-130259

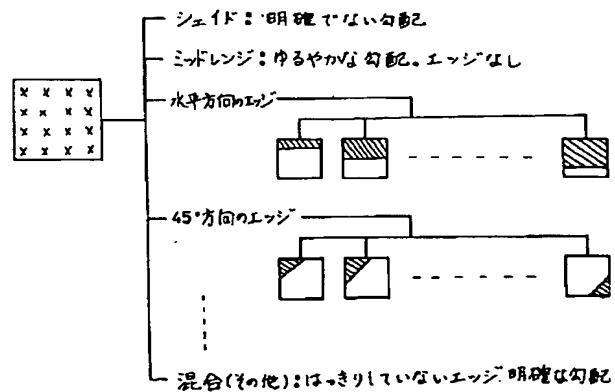
16

\* 107 出力ベクトルコードテーブルメモリ  
603 コードブック出力ベクトルレジスタ

\*

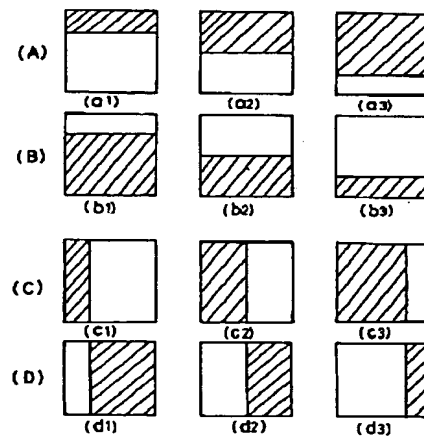
【図2】

ブロックのクラス分けを説明する図



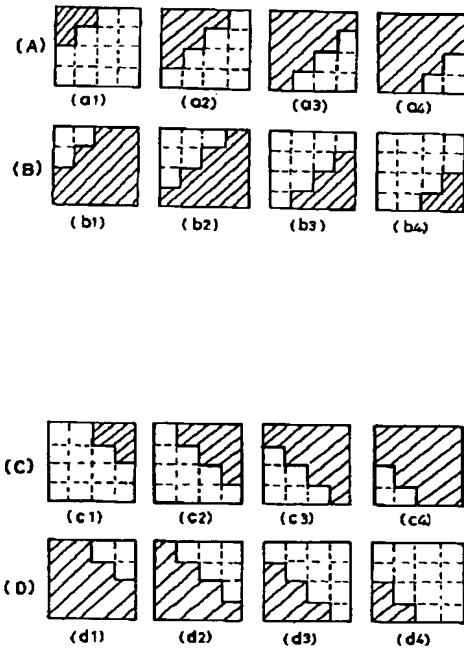
【図4】

エッジブロックを示す図(その1)



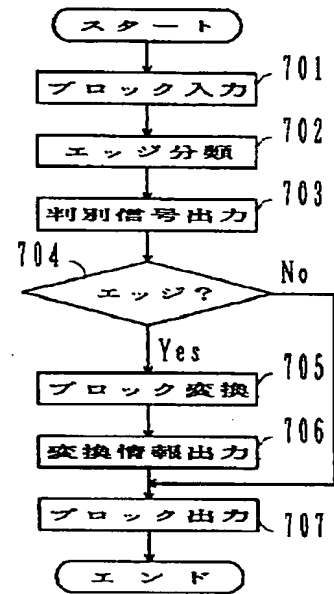
【図5】

エッジロックを示す図(その2)



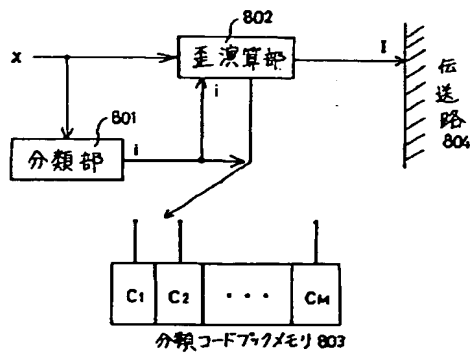
【図7】

分類処理の動作フローチャート



【図8】

従来のベクトル量子化符号化装置のブロック図



【図6】

歪演算部103の構成ブロック図

